Postopek množenja velikih števil (Karatsuba, Ofman /1962)

Problem :
v okviru računalniškega sistema smo vedno omejeni s pomnilnikov, s tem pa so omejene tudi podatkovne strukture, ki jih uporabljamo za hranjenje vrednosti / podatkov. Tipično se velikost podatkovne strukture izbere pri načrtovanju centralne procesne enote in je kategorija, ki je ekonomsko opredeljena. Zakaj ekonomsko ?
1. Podatki se procesirajo znotraj CPE, podatek je med procesiranju v registru CPE (načelno je to res, čeprav poznamo RS brez registrov). Register je pomnilnik, narejen v tehnologiji CPE. Večji kot je dražji je.
2. Kako velike podatke pa dejansko potrebujemo pri vsakodnevnem procesiranju ? Analiza pokaže, da velikost ni prav velika. V veliki večini primerov premetavamo številke, katerih velikost enostavno spravimo v 2 zloga pomnilnika, če ne v dva, pa v štiri gotovo. Če za podatek rezerviramo 4 zloge, pa dejansko uporabljamo le 2 od teh štirih, dodatna dva zloga brez pomena prenašamo po vodilih (ozko grlo VonNeumann). V parih % rabimo večji podatek, nekateri uporabniki nikoli.
Ekonomska računica je jasna : poceni (dovolj kratki podatki), ker tako več prodaš in s tem več zaslužiš; tisti »profiji«., ki rabijo več, pa naj plačajo več za ustreznejši RS , ali pa naj čakajo, da se bo 4x daljši podatek k CPE prenašal 4x dalj časa in da bo tudi procesiranje malo dalj trajalo.

Ker smo profiji, pa brez para, se bomo ukvarjali s slednjim. Zato si najprej oglejmo, kako se izvaja postopek množenja, kot smo se ga učili v šoli (za demonstracijo bodo izbrana 'rahlo' manjša števila):

```
86x42 =
344
  172
-------
 3612
```

Ali če množimo, kot da bi množili dvočlenike :

```
86 x 42 =  (8*10 + 6) x (4*10 + 2) =
                  8*4*10*10  + 6*4*10  + 8*2*10 + 6*2
```

Če množenje z 10 smatramo za (decimalen) pomik števila v levo, potem lahko ugotovimo, da smo za izračun slednjega rezultata porabili 4 množenja in 3 seštevanja.
Očitno je število množenj kvadratično odvisno od števila števk v množencih,

Cilj je : pohitriti postopek oz. zmanjšati število časovno najbolj potratnih operacij; to je množenj. Predhodni postopek bi lahko posplošili takole :

```
AB x CD =  (A*10 + B) x (C*10 + D) =
            A*C*10*10   +   B*C*10   +   A*D*10  +  B*D
```

Programski postopek : produkti : A*C, B*C, A*D, B*D
desetiški zamiki : 10^2, 10, 1
izračun vsot

če malo premečemo formulo :
$(A*10 + B) \times (C*10 + D) = A*C*10^2 + [(A+B)*(C+D)*10-AC-BD] + BD$

Lahko vidimo, da se v končnem izrazu pojavljajo le še trije zmnožki :
AC, BD in (A+B)*(C+D)

Da ne bi preveč kompliciral, kompleksnost se zmanjša iz $\mathbf{n^2}$ na približno $\mathbf{n^{1.58}}$
Teorijo si lahko ogledate na : www.ccas.ru/personal/karatsuba/divce.htm

Postopek :
Ker velikih števil ne moremo procesirati, jih bomo morali najprej razdeliti in jih deliti toliko časa, dokler ne bo problem (zaradi velikosti podatkov obvladljiv), obvladljive dele bomo sprocesirali. Da pa bi dobili končni rezultat, bo potrebno vanj združiti vse dobljene delne rezultate. Tako reševanje, oz. postopke, ki jih rešujemo na tak način, uvrščamo v kategorijo postopkov, poimenovano deli in vladaj (divide et impera, divide and conquer). Z nekaj postopki iz te kategorije smo že imeli opravka (binarna delitev pri binarnem skanju, hitro razvrščanje, …)

Primer : recimo, da je obvladljivo le množenje enomestnih števil

```
4 3 2 1 * 1 2 3 4 =
  A  B     C  D

Produkt 1 : A*C =  4 3 * 1 2  (neobvladljiv, zato razcep)
                   A B   C D

                   P11 : A*C = 4*1 = 4 (obvladljivo)
                   P12 : B*D = 3*2 = 6
                   P13 : (A+B)*(C+D) = 7*3 = 21

                 = P11*100 + P12 + (P13 – P11 – P12 )*10 (sestavimo v delni rezultat)
                 = 400 + 6 + 110 = 516

Produkt 2 : B*D =   2 1 * 3 4 (neobvladljiv)
                    A B   C D

                    P21 : A*C = 2*3 = 6
                    P22 : B*D = 1*4 = 4
                    P23 : (A+B)*(C+D) = 3*7 = 21

                 =  P21*100 + P22 + P23*10
                 =  600 + 4 + 110 = 714

Produkt 3 : (A+B)*(C+D) =  6 4  *  4 6   (neobvladljiv)
                           A B     C D

                           P31 = A*C = 6*4 = 24
                           P32 = B*D = 4*6 = 24
                           P33 = (A+B)*(C+D) = 1 0  * 1 0 (neobvladljivo)
                                               A B    C D
                                               P331 = 1
                                               P332 = 0
                                               P333 = 0
                                             = 1*100 + 0 + 0*10 = 100
                                 = 100

                          = 24*100 + 24 + 52*10 = 2944

Rezultat = 516*10000 + 714 + 1714*100 = 5332114
```